(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 553 778 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
13.07.2005 Bulletin 2005/28

(51) Int Cl.[7]: **H04N 7/26**, G06T 9/00, H04N 7/30, G06T 5/00

(21) Application number: **04030974.2**

(22) Date of filing: **29.12.2004**

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IS IT LI LT LU MC NL PL PT RO SE SI SK TR
Designated Extension States:
AL BA HR LV MK YU

(30) Priority: **06.01.2004 US 535045 P**
06.01.2004 US 535050 P
31.03.2004 US 815029

(71) Applicant: **Sharp Kabushiki Kaisha**
Osaka-shi, Osaka 545-8522 (JP)

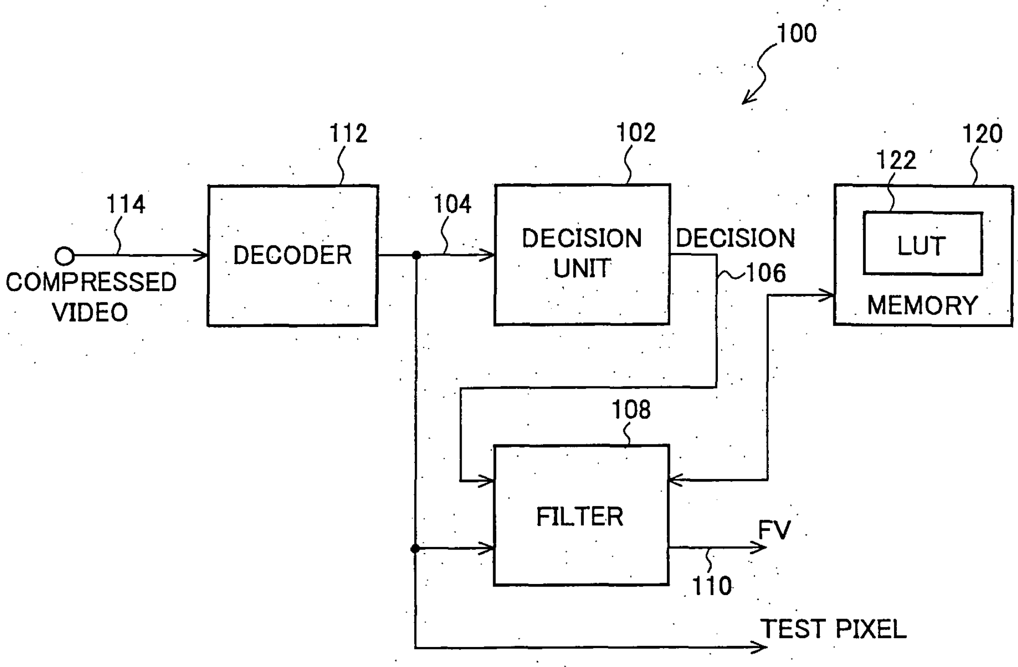
(72) Inventor: **Deshpande, Sachin Govind**
Vancouver WA 98683, Clark (US)

(74) Representative: **Müller - Hoffmann & Partner**
Patentanwälte,
Innere Wiener Strasse 17
81667 München (DE)

(54) De-ringing filter

(57) An image de-ringing filter system and method are provided. The method comprises: accepting image pixels (504); collecting data from a first group of pixels neighboring a test pixel (506); in response to the first group data, deciding if the test pixel includes image ringing artifacts (508); collecting data from a second group of pixels neighboring the test pixel (510); in response to the second group data, generating a filtered value (FV) (512b/512f); and, replacing the test pixel actual value with FV (514). Typically, collecting data from the first and second group of pixels includes the performance of a mathematical operation. For example, a matrix may be defined for the multiplication of the first group of pixels. Values of pixels on a first side of the coordinate axis may be subtracted from pixels on a second side of the coordinate axis, opposite of the first side. Then, the difference is compared to a threshold.

FIG.1



EP 1 553 778 A1

Description

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0001] This invention generally relates to compressed image and video coding and, more particularly, to a method for filtering ringing artifacts that may occur as a result of compressed image and video encoding/decoding processes.

2. Description of the Related Art

[0002] Computation resources and bandwidth can be saved by encoding images and videos at a low bit-rate. However, low bit-rate encoding may result in several types of artifacts in the decompressed images. The most notable artifacts include blocking and ringing artifacts. The ringing artifacts are typically observed around the true edges of an image. The ringing artifacts are also referred to as mosquito artifacts, as they tend to be annoying, especially in moving images (video sequences). A variety of filters exist for filtering out these unwanted artifacts. These include de-blocking and de-ringing filters. For de-ringing, conventional methods operate in both the transform and pixel domains. Other conventional de-ringing filters make use of quantization information. One drawback of all the above-mentioned de-ringing filters is that are computationally intensive. Thus, the filters are not suitable for all receiving systems. Further, the filters may result in unacceptable delays, even when they can be implemented.

[0003] It would be advantageous if a low complexity de-ringing filter could be developed for ringing artifact reduction.

SUMMARY OF THE INVENTION

[0004] The present invention is a de-ringing filter with a low computational complexity. A decision to apply the filter is made for each pixel based on its edge strength. In one aspect, a 3x3 kernel is used for filtering. Only the non-edge neighbor pixels are used to filter the current (test) pixel. In this aspect, the filter uses all of the non-edge neighbor pixels and the current pixel weighted appropriately, based on the total number of non-edge neighbor pixels. The invention works entirely in the pixel domain and does not use or need any quantization information. Further, the solution is not necessarily block or macroblock-based.

[0005] Accordingly, an image de-ringing filter method is provided. The method comprises: accepting a plurality of image pixels; collecting data from a first group of pixels neighboring a test pixel; in response to the first group data, deciding if the test pixel includes image ringing artifacts; collecting data from a second group of pixels neighboring the test pixel; in response to the second group data, generating a filtered value (FV); and, replacing the test pixel actual value with FV.

[0006] Typically, collecting data from the first and second group of pixels includes the performance of a mathematical operation. For example, a matrix may be defined for the multiplication of the first group of pixels. The mathematical operation may involve the comparison of pixels values on opposite sides of a coordinate axis bisecting the test pixel. More specifically, values of pixels on a first side of the coordinate axis may be subtracted from pixels on a second side of the coordinate axis, opposite of the first side. Then, the difference is compared to a threshold.

[0007] In another aspect, generating a FV in response to the second group operation includes: generating a map value for each pixel in the second group; and, using pixels from the second group to calculate FV, if they are equal to a first map value. Specifics of map values and the definition of the second group are provided.

[0008] Additional details of the above-described method, and an image de-ringing filter system are provided below.

BRIEF DESCRIPTION OF THE DRAWINGS

[0009]

Fig. 1 is a schematic block diagram of the present invention image de-ringing filter system.

Fig. 2 is a diagram depicting a test pixel, and a group of neighboring pixels.

Fig. 3 is a drawing depicting the LUT of Fig. 1.

Fig. 4 is a drawing illustrating an exemplary aspect of the present invention filter.

Fig. 5 is a flowchart illustrating the present invention image de-ringing filter method.

Fig. 6 is a flowchart illustrating another aspect of the present invention image de-ringing filter method.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0010] Fig. 1 is a schematic block diagram of the present invention image de-ringing filter system. The system 100 comprises a decision unit 102 having an input on line 104 to accept a plurality of image pixels. The decision unit 102 collects data from a first group of pixels neighboring a test pixel. In response to the first group data, the decision unit 102 supplies a decision at an output on line 106 as to whether the test pixel includes image ringing artifacts. A filter 108 has an input on line 104 to accept the plurality of image pixels and an input on line 106 to accept the decision. The filter 108 collects data from a second group of pixels neighboring the test pixel. In response to the second group data, the filter generates a filtered value (FV) and supplies the FV at an output on line 110 as a replacement to the test pixel actual value.

[0011] A decoder 112 has a connection on line 114 to accepted encoded video information. The encoded video may come from a network, local storage, or other source. This information may be encoded in a standard such as moving picture experts group (MPEG) or H.264 standards, to name a few examples. The decoder 112 has an output on line 104 to supply the plurality of image pixels to the decision unit 102, and to the filter 108, as decoded image information.

[0012] Fig. 2 is a diagram depicting a test pixel, and a group of neighboring pixels. Pixels from the first group are represented with a "1", while pixels from the second group are represented with a "2". Note, the first and second group of pixels are not necessarily the same. Although 9 pixel positions are shown, neither the first nor the second group of pixels is limited to any particular number.

[0013] Typically, the filter performs a mathematical operation on the second group of pixels. Likewise, the decision unit typically performs a mathematical operation on the first group of pixels. For example, the decision unit may define a matrix and multiply the first group of pixels by the matrix, or more than one matrix. In some aspects, the decision unit defines a matrix such that a zero value is assigned to the position of the test pixel in the matrix. In some aspects matrix may be used for a vector dot product. In one aspect, the decision unit may compare values of pixels on opposite sides of a coordinate axis bisecting the test pixel. For example, 1a, 1d, and 1g may be compared to 1c, 1f, and 1i. In another aspect, the decision unit subtracts the values of pixels on a first side of the coordinate axis from pixels on a second side of the coordinate axis, opposite of the first side, and compares the difference to a threshold.

[0014] In a different aspect, the decision unit compares the values of pixels on opposite sides of a plurality of coordinate axes, oriented in a plurality of directions. For example, pixels 1a and 1c may be compared to pixels 1g and 1i, while pixels 1a and 1g are compared to pixels 1c and 1i. In one aspect, the decision unit collects data from a group of 4 pixels neighboring the test pixel. For example, pixel positions 1a, 1c, 1g, and 1i can be used. The results of any of the mathematical operations can be compared to a threshold as a means of making a decision as to whether a test pixel is to be filtered.

[0015] With respect to a test pixel P(i,j), with i and j indicating row and column indices, respectively, and P (i,j) representing a pixel gray value, operators H1 and H2 may be used to derive gradient values $g_{H1}(i,j)$ and $g_{H2}(i,j)$, respectively, where

$$H1 = [10\ -1];$$

and,

$$H2=\begin{bmatrix}1\\0\\-1\end{bmatrix}$$

[0016] The decision unit calculates $S(i,j) = (|g_{H1}(i,j)| + |g_{H2}(i,j)| + 1) >> 1$, where >> x represents a binary value right-shift of x. Then, the decision unit decides that P(i, j) is a ringing artifact, if S(i,j) < threshold. Note, this examples uses 4 test pixel neighbors, from the immediately neighboring 8 pixels to make a filter decision.

[0017] In other aspects, the position of pixels, the number of pixels, and the size of the neighboring group from which the pixels are selected may be different. For example, the operators may be as follows:

$$H1 = [110\ -1 -1];$$

and,

$$H2=\begin{bmatrix}1\\1\\0\\-1\\-1\end{bmatrix}$$

[0018] X-axis, reflection, y-axis reflection, diagonal reflection, diagonal reflection symmetry, -90 degree rotation symmetry, centro symmetry, quadrantal symmetry, diagonal symmetry, 4-fold rotation symmetry, and octagonal symmetry are examples of other coordinate axes comparisons that can be made.

[0019] Referring now to the filter, in some aspects the filter may add the test pixel to the second group of pixels. Alternately, the test pixel value is not used to calculate FV. In one aspect, the filter collects data from 8 pixels neighboring the test pixel (2a, 2b, 2c, 2d, 2f, 2g, 2h, and 2i). However, other definitions of the second pixel group are possible.

[0020] The decision unit, in response to comparing S (i,j) to the threshold, generates a map value M(i,j) for P (i,j), where:

$$M(i,j) = 1, \text{ if } S(i,j) \geq \text{threshold};$$

and,

$$M(i,j) = 0, \text{ if } S(i,j) < \text{threshold};$$

[0021] Then, the filter uses pixels from the second group to calculate FV, if they are equal to a first map value. In some aspects, the filter uses a first map value equal to 0. Alternately, the filter uses a first map value not equal to 1.

[0022] In one aspect of the system, the filter selects pixels from the second group to calculate FV, if they are equal to the first map value. For example, the filter may randomly select pixels from the second group to calculate FV, if they are equal to the first map value. More generally, the filter may accept a plurality of image pixel sets, in a plurality of frames. Then, it generates FV by randomly selecting a first collection of pixel positions with respect to the test pixel, and uses pixels in the first collection to calculate FV for each test pixel in every image pixel set, in every frame.

[0023] In another aspect, the filter generates FV by randomly selecting a first collection of pixel positions with respect to the test pixel in a first image pixel set in a current frame, and uses pixels in the first collection to calculate FV for each test pixel in every image pixel set in the current frame. Further, the filter randomly reselects a second collection of pixel positions in an image pixel set in a frame subsequent to the current frame, and uses pixels in the second collection to calculate FV for each test pixel in every image pixel set in the subsequent frame.

[0024] In another aspect, the filter selects pixels in predetermined pixel positions, with respect to the test pixel, from the second group to calculate FV, if it is equal to the first map value. More generally, the filter may accept a plurality of image pixel sets in a plurality of frames, and select the pixels in the predetermined pixel positions to calculate FV for each test pixel in every image pixel set, in every frame. For example, the filter may select the pixels in a predetermined first collection of pixel positions to calculate FV for each test pixel in every image pixel set in a current frame, and select the pixels in a predetermined second collection of pixel positions to calculate FV for each test pixel in every image pixel set in a frame subsequent to the current frame.

[0025] In another aspect, the filter generates FV by selecting the pixels in the predetermined first collection of pixel positions to calculate FV for test pixels in a first image pixel set and, then, selecting the pixels in the predetermined second collection of pixel positions to calculate FV for test pixels in a second image pixel set.

[0026] In a different aspect of the invention, the filter uses pixels from the second group to calculate FV, if they are equal to a first map value, by selectively weighting second group pixel values. Then, the weighted values are summed and averaged. In this aspect, the filter may add the test pixel to the second group of pixels. The filter may also selectively weigh in response to number of pixels in the second group.

[0027] The following is a specific example of a filter algorithm. The filter generates FV by:

calculating nV = sum of second group pixel values for pixels having a map value of 0;
calculating nE = total number of pixels in the second group with a map value of 0;
if nE = 1, then

$$FV = (nV + P(i,j) + 1) >> 1;$$

else, if nE < 4, then

$$nV = nV + (4 - nE) * P(i,j);$$

and,

$$FV = (nV + 2) >> 2;$$

else, if nE < 8, then

$$nV = nV + (8 - nE) * P(i,j);$$

and,

$$FV = (nV + 4) >> 3;$$

else, if nE = 8, then

$$nV = nV - P(i + 1, j + 1) + P(i,j);$$

and,

$$FV = (nV + 4) >> 3.$$

[0028] In an alternate algorithm, the filter generates FV by:

calculating nV = sum of second group pixel values for pixels having a map value of 0;
calculating nE = total number pixels in the second group with a map value of 0;
if nE = 1, then

$$FV = (nV + P(i,j) + 1) >> 1;$$

else, if nE < 4, then

$$nV = nV + (4 - nE) * P(ij);$$

and,

$$FV = (nV) >> 2;$$

else, if nE < 8, then

$$nV = nV + (8 - nE) * P(i,j);$$

and,

$$FV = (nV) >> 3;$$

else, if nE = 8, then

$$nV = nV - P(i + 1, j + 1) + P(i,j);$$

and,

$$FV = (nV) >> 3.$$

[0029] Returning to Fig. 1, some aspects of the system 100 include an accessible memory 120 including a lookup table (LUT) 122 with pre-calculated values. The filter 108 generates a FV in response accessing the LUT 122.

[0030] Fig. 3 is a drawing depicting the LUT of Fig. 1. As shown, the LUT 122 is indexed by nE values. The filter calculates nE, where nE = the total number of pixels in the second group with the first map value. The filter then uses the calculated nE value to access the LUT 122. In one aspect, the LUT 122 includes a value for each nE indicating the number of times the test pixel P (i,j) is added. In another aspect, the LUT 122 includes a value for each nE indicating the number of times the result is right shifted. In a different aspect, the LUT 122 includes a value for each nE indicating if a pixel value from second group of pixels is subtracted, or not.

Functional Description

[0031] As described above, the present invention de-ringing filter consists of decision and filtering functions. The following description is one specific example of the de-ringing filter.

Decision Stage

[0032] For each pixel, a decision is made as to whether the pixel should be filtered or left unprocessed. The decision is based on following computation:

[0033] For each pixel, use the operators H_1 = [1 0 -1] and

$$H_2 = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

to compute two gradient values that are denoted by $g_{H1}(i, j)$ and $g_{H2}(i, j)$. The operators are selected so that the actual current (test) pixel itself is not used in the calculation of its *Strength(i, j)* . Instead, only its 4 neighbors are used for this computation. Then a local measure *Strength*(*i*, *j*) for the current pixel (*i*, *j*) is calculated as:

$$Strength(i,j) = (| g_{H_1}(i, j) | + | g_{H_2}(i, j) 1+1) >> 1;$$

$$\text{if } (Strength(i, j) >= Threshold)$$
{

$$Map(i, j) = 1;$$

}
else
{

$$Map(i, j) = 0;$$

Apply De-ringing filter;

}

,where (*i*, *j*) is the pixel index and *Threshold* is a parameter which controls the filtering decision. The *Threshold* is a parameter that can be manipulated to vary the effective strength of the filter. A high-threshold value results in more pixels getting filtered and, thus, a stronger filter. A low-threshold value results in a lesser number of pixels getting filtered and, thus, a weaker filter.

[0034] If the *Strength*(*i, j*) of a test pixel is less than the threshold, then the de-ringing filter is applied to this pixel. Otherwise, the pixel is left unprocessed, i.e., its value is used without changing. If the decision is made to apply a de-ringing filter to a pixel, then the *Strength* (*i, j*) value is computed and compared to the *Threshold* , to obtain the *Map* (*i*, *j*) values for the current pixel's 8 neighbors in the 3x3 size kernel. These values are used in the de-ringing filter stage, as explained below.

Low complexity adaptive filtering stage

[0035] Based on the decision from the previous stage, the pixels are processed by de-ringing filter, or not processed. The filter has a low complexity, as compared to other approaches, and is signal adaptive. The filtering can be done in place, or the results could be stored in a separate memory store. For example, satisfactory results are obtained with an in-place computation approach, using a 3x3 kernel.

[0036] Fig. 4 is a drawing illustrating an exemplary aspect of the present invention filter. The filter can be realized using only addition, subtraction, and shift operations. The multiplication operations in Fig. 4 can also be realized as multiple addition operations. In one case, a randomly (or selectively - based on some criterion) chosen neighbor pixel is not used in the filtering of the current pixel (if nE is equal to 8, for example). A lookup table can be used to realize the filter. The lookup table can be indexed by the value nE, and can store the information about the number of times the center pixel is added. The LUT can also be used if any neighbor pixels need to be subtracted (or effectively not used). The resulting average value is used to replace the current pixel under consideration.

[0037] Fig. 5 is a flowchart illustrating the present invention image de-ringing filter method. Although the method is depicted as a sequence of numbered steps for clarity, no order should be inferred from the numbering unless explicitly stated. It should be understood that some of these steps may be skipped, performed in parallel, or performed without the requirement of maintaining a strict order of sequence. The method starts at Step 500.

[0038] Step 502 decodes compressed image information. Step 504 accepts a plurality of image pixels. That is, the decoded image information is accepted. Step 506 collects data from a first group of pixels neighboring a test pixel. Step 508 decides if the test pixel includes image ringing artifacts, in response to the first group data. Step 510 collects data from a second group of pixels neighboring the test pixel. Step 512, in response to the second group data, generates a filtered value (FV). Step 514 replaces the test pixel actual value with FV. Note, FV is not necessarily calculated, or used to replace the test pixel, depending upon the result of the decision process in Step 508.

[0039] In one aspect, collecting data from a second group of pixels neighboring a test pixel in Step 510 includes performing a mathematical operation on the second group of pixels. For example, collecting data from a second group of pixels neighboring the test pixel in Step 510 includes collecting data from 8 pixels neighboring the test pixel. In another aspect, Step 510 adds the test pixel to the second group of pixels.

[0040] Likewise, collecting data from a first group of pixels neighboring a test pixel in Step 506 may include performing a mathematical operation on the first group of pixels. In some aspects, Step 506 compares the results of the mathematical operation to a threshold.

[0041] For example, performing a mathematical operation on the first group of pixels may include substeps. Step 506a defines a matrix. Step 506b multiplies the first group of pixels by the matrix. In one aspect, the matrix is defined such that a zero value is assigned to the position of the test pixel in the matrix.

[0042] In one aspect, performing a mathematical operation on the first group of pixels (Step 506) may include comparing values of pixels on opposite sides of a coordinate axis bisecting the test pixel. For example, comparing values of pixels on opposite sides of a coordinate axis bisecting the test pixel may include: subtracting the values of pixels on a first side of the coordinate axis from pixels on a second side of the coordinate axis, opposite of the first side; and, comparing the difference to a threshold. In some aspects, a fixed threshold value is selected. Further, the values of pixels may be compared on opposite sides of a plurality of coordinate axes, oriented in a plurality of directions. In another example, data is collected from a group of 4 pixels neighboring the test pixel.

[0043] More specifically, values of pixels on opposite sides of a coordinate axis bisecting the test pixel may be compared (Step 506) as follows:

with respect to a test pixel P(i,j), with i and j indicating row and column indices, respectively, and P(i,j) representing a pixel gray value, using operators H1 and H2 to derive gradient values gH1(i,j) and g_{H2}(i, j), respectively, where

$$H1 = [10\ -1];$$

and,

$$H2=\begin{bmatrix}1\\0\\-1\end{bmatrix}$$

calculating $S(i,j) = (|g_{H1}(i,j)| + |g_{H2}(i,j)| + 1) >> 1$;

where >> x represents a binary value right-shift of x.

[0044] Then, deciding if the test pixel includes image ringing artifacts in Step 508 includes deciding that P(i,j) includes ringing artifacts, if S(i,j) < threshold.

[0045] Additionally, Step 506 may, in response to comparing S(i,j) to the threshold, generate a map value M(i,j) for P(i,j), where:

$$M(i,j) = 1, \text{ if } S(i,j) \geq \text{threshold};$$

and,

$$M(i,j) = 0, \text{ if } S(ij) < \text{threshold}.$$

[0046] Then, generating FV in Step 512 includes using pixels from the second group to calculate FV, if they are equal to a first map value. In some aspects, FV is generated using a first map value of 0. In other aspects, the first map value is not equal to 1. Alternately, FV can be calculated by selecting pixels from the second group, if they are equal to the first map value.

[0047] For example, pixels may be randomly selected from the second group for the calculation of FV, if they are equal to the first map value. If Step 504 accepts a plurality of image pixel sets, in a plurality of frames, then generating FV in Step 512 may include substeps. Step 512a randomly selects a first collection of pixel positions with respect to the test pixel. Step 512b uses pixels in the first collection to calculate FV for each test pixel in every image pixel set, in every frame. In a different aspect, Step 512a randomly selects a first collection of pixel positions with respect to the test pixel in a first image pixel set in a current frame. Step 512b uses pixels in the first collection to calculate FV for each test pixel in every image pixel set in the current frame. Step 512c (not shown) randomly reselects a second collection of pixel positions in an image pixel set in a frame subsequent to the current frame. Step 512d (not shown) uses pixels in the second collection to calculate FV for each test pixel in every image pixel set in the subsequent frame.

[0048] In another aspect, Step 512 selects a predetermined collection of pixel positions with respect to the test pixel. If Step 504 accepts a plurality of image pixel sets in a plurality of frames, then Step 512 generates FV by selecting the pixels in the predetermined pixel positions to calculate FV for each test pixel in every image pixel set, in every frame. Alternately, Step 512e selects the pixels in a predetermined first collection of pixel positions to calculate FV for each test pixel in every image pixel set in a current frame. Step 512f selects the pixels in a predetermined second collection of pixel positions to calculate FV for each test pixel in every image pixel set in a frame subsequent to the current frame. As another alternative, Step 512e may select the pixels in the predetermined first collection of pixel positions to calculate FV for test pixels in a first image pixel set. Then, Step 512f selects the pixels in the predetermined second collection of pixel positions to calculate FV for test pixels in a second image pixel set.

[0049] In another aspect, using pixels from the second group to calculate FV, if they are equal to a first map value (Step 512), may include other substeps (not shown). Step 512g selectively weights second group pixel values. Step 512h sums the weighted values. Step 512i averages. With this aspect, the test pixel may be added to the second group of pixels. Further, Step 512g may weigh the pixels in response to the number of pixels in the second group.

[0050] In another aspect, the generation of FV (Step 512) may include:

calculating nV = sum of second group pixel values for pixels having a map value of 0;
calculating nE = total number of pixels in the second group with a map value of 0;
if nE = 1, then

$$FV = (nV + P(i,j) + 1) >> 1;$$

else, if nE < 4, then

$$nV = nV + (4 - nE) * P(i,j);$$

and,

$$FV = (nV + 2) >> 2;$$

else, if nE < 8, then

$$nV = nV + (8 - nE) * P(i,j);$$

and,

$$FV = (nV + 4) >> 3;$$

else, if nE = 8, then

$$nV = nV - P(i + 1, j + 1) + P(i,j);$$

and,

$$FV = (nV + 4) >> 3.$$

[0051] Alternately, the generation of FV may include:

calculating nV = sum of second group pixel values for pixels having a map value of 0;
calculating nE = total number pixels in the second group with a map value of 0;
if nE = 1, then

$$FV = (nV + P(i,j) + 1) >> 1;$$

else, if nE < 4, then

$$nV = nV + (4 - nE) * P(i,j);$$

and,

$$FV = (nV) >> 2;$$

else, if nE < 8, then

$$nV = nV + (8 - nE) * P(i,j);$$

and,

$$FV = (nV) >> 3;$$

else, if nE = 8, then

$$nV = nV - P(i + 1, j + 1) + P(i,j);$$

and,

$$FV = (nV) >> 3.$$

[0052] In other aspects, generating FV in response to the second group data includes other substeps (not shown). Step 512j loads a lookup table (LUT) with the pre-calculated values. Typically, the LUT is loaded before the decision and filtering processes are performed. Step 512k accesses the LUT. For example, Step 512k1 calculates nE = the total number of pixels in the second group with the first map value. Then, Step 512k2 uses nE to access the LUT.

[0053] In one aspect, Step 512j loads a value for each nE indicating the number of times the test pixel P(i,j) is added. In another aspect, Step 512j loads a value for each nE indicating the number of times the result is right shifted. In a third aspect, Step 512j loads a value for each nE indicating if a pixel value from second group of pixels is subtracted or not.

[0054] Fig. 6 is a flowchart illustrating another aspect of the present invention image de-ringing filter method. The method starts at Step 600. Step 602 accepts a plurality of image pixels. Step 604 performs a mathematical operation on a first group of pixels neighboring a test pixel. Step 606, in response to the first group operation, decides if the test pixel includes image ringing artifacts. Step 608 performs a mathematical operation on a second group of pixels neighboring the test pixel. Step 610, in response to the second group operation, generates FV. Step 612 replaces the test pixel actual value with FV.

[0055] In one aspect, performing a mathematical operation on the first group of pixels (Step 604) includes: defining a matrix; and, multiplying the first group of pixels by the matrix. In another aspect, Step 604 compares values of pixels on opposite sides of a coordinate axis bisecting the test pixel.

[0056] In one aspect, generating a FV in response to the second group operation (Step 610) includes: generating a map value for each pixel in the second group; and, using pixels from the second group to calculate FV, if they are equal to a first map value.

[0057] A system and method have been provided for removing ringing artifacts that can be simply implemented after a compressed video decoding process. Some examples of specific algorithms have been described to clarify the invention. However, the invention is not limited to merely these examples. Although abstract compressed video standards have been described, the present invention may be adapted for use with the following video standards: MPEG1, MPEG2, MPEG4, H.263, H.263+, H.263++, and H.264. Other variations and embodiments of the invention will occur to those skilled in the art.

Claims

1. An image de-ringing filter method, the method comprising:

   accepting a plurality of image pixels;
   collecting data from a first group of pixels neighboring a test pixel;
   in response to the first group data, deciding if the test pixel includes image ringing artifacts;
   collecting data from a second group of pixels neighboring the test pixel;
   in response to the second group data, generating a filtered value (FV); and,
   replacing the test pixel actual value with FV.

2. The method of claim 1 wherein collecting data from a second group of pixels neighboring a test pixel includes performing a mathematical operation on the second group of pixels.

3. The method of claim 1 wherein collecting data from a first group of pixels neighboring a test pixel includes performing a mathematical operation on the first group of pixels.

4. The method of claim 3 wherein collecting data from a first group of pixels neighboring a test pixel further includes comparing the results of the mathematical operation to a threshold.

5. The method of claim 3 wherein performing a mathematical operation on the first group of pixels includes:

   defining a matrix; and,
   multiplying the first group of pixels by the matrix.

6. The method of claim 5 wherein the matrix is defined such that a zero value is assigned to the position of

the test pixel in the matrix.

7. The method of claim 3 wherein performing a mathematical operation on the first group of pixels includes comparing values of pixels on opposite sides of a coordinate axis bisecting the test pixel.

8. The method of claim 7 wherein comparing values of pixels on opposite sides of a coordinate axis bisecting the test pixel includes:

subtracting the values of pixels on a first side of the coordinate axis from pixels on a second side of the coordinate axis, opposite of the first side; and,
comparing the difference to a threshold.

9. The method of claim 8 wherein comparing values of pixels on opposite sides of a coordinate axis bisecting the test pixel includes comparing the values of pixels on opposite sides of a plurality of coordinate axes, oriented in a plurality of directions.

10. The method of claim 7 wherein collecting data from a first group of pixels neighboring a test pixel includes collecting data from a group of 4 pixels neighboring the test pixel.

11. The method of claim 9 wherein comparing values of pixels on opposite sides of a coordinate axis bisecting the test pixel includes:

with respect to a test pixel P(i,j), with i and j indicating row and column indices, respectively, and P(i,j) representing a pixel gray value, using operators H1 and H2 to derive gradient values $g_{H1}(i,j)$ and $g_{H2}(i,j)$, respectively, where

$$H1 = [1\ 0\ -1];$$

and,

$$H2 = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

calculating

$$S(i,j) = (|g_{H1}(i,j)| + |g_{H2}(i,j)| + 1) >> 1;$$

where >> x represents a binary value right-shift of x;

wherein deciding if the test pixel includes image ringing artifacts includes deciding that P(ij) includes ringing artifacts, if S(i,j) < threshold.

12. The method of claim 7 further comprising:

selecting a fixed threshold value.

13. The method of claim 1 further comprising:

decoding compressed image information; and,

wherein accepting a plurality of image pixels includes accepting the decoded image information.

14. The method of claim 1 wherein collecting data from a second group of pixels neighboring the test pixel includes adding the test pixel to the second group of pixels.

15. The method of claim 1 wherein collecting data from a second group of pixels neighboring the test pixel includes collecting data from 8 pixels neighboring the test pixel.

16. The method of claim 11 wherein comparing values of pixels on opposite sides of a coordinate axis bisecting the test pixel additional includes:

in response to comparing S(i,j) to the threshold, generating a map value M(i,j) for P(i,j), where:

$$M(i,j) = 1, \text{ if } S(i,j) \geq \text{threshold};$$

and,

$$M(i,j) = 0, \text{ if } S(i,j) < \text{threshold};$$

and,

wherein generating FV includes using pixels from the second group to calculate FV, if they are equal to a first map value.

17. The method of claim 16 wherein generating FV includes the first map value being equal to 0.

18. The method of claim 16 wherein generating FV includes the first map value not being equal to 1.

19. The method of claim 16 wherein generating FV includes selecting pixels from the second group to calculate FV, if they are equal to the first map value.

20. The method of claim 19 wherein generating FV includes randomly selecting pixels from the second group to calculate FV, if they are equal to the first map value.

21. The method of claim 20 wherein accepting a plurality of image pixels includes accepting a plurality of image pixel sets, in a plurality of frames;
wherein generating FV includes:

randomly selecting pixels in a first collection of pixel positions with respect to the test pixel;
using pixels in the first collection to calculate FV for each test pixel in every image pixel set, in every frame.

22. The method of claim 20 wherein accepting a plurality of image pixels includes accepting a plurality of image pixel sets, in a plurality of frames;
wherein generating FV includes:

randomly selecting a first collection of pixel positions with respect to the test pixel in a first image pixel set in a current frame;
using pixels in the first collection to calculate FV for each test pixel in every image pixel set in the current frame;
randomly reselecting a second collection of pixel positions in an image pixel set in a frame subsequent to the current frame; and,
using pixels in the second collection to calculate FV for each test pixel in every image pixel set in the subsequent frame.

23. The method of claim 19 wherein selecting pixels from the second group to calculate FV, if they are equal to the first map value, includes selecting a predetermined collection of pixel positions with respect to the test pixel.

24. The method of claim 23 wherein accepting a plurality of image pixels includes accepting a plurality of image pixel sets in a plurality of frames;
wherein generating FV includes selecting the pixels in a predetermined first collection of pixel positions to calculate FV for each test pixel in every image pixel set, in every frame.

25. The method of claim 23 wherein accepting a plurality of image pixels includes accepting a plurality of image pixel sets in a plurality of frames;
wherein generating FV includes:

selecting the pixels in a predetermined first collection of pixel positions to calculate FV for each test pixel in every image pixel set in a current frame; and,
selecting the pixels in a predetermined second collection of pixel positions to calculate FV for each test pixel in every image pixel set in a frame subsequent to the current frame.

26. The method of claim 23 wherein generating FV includes:

selecting the pixels in a predetermined first collection of pixel positions to calculate FV for each test pixel in a first image pixel set; and,
selecting the pixels in a predetermined second collection of pixel positions to calculate FV for each test pixel in a second image pixel set.

27. The method of claim 16 wherein using pixels from the second group to calculate FV, if they are equal to a first map value, includes:

selectively weighting second group pixel values;
summing the weighted values; and,
averaging.

28. The method of claim 27 wherein collecting data from a second group of pixels neighboring the test pixel includes adding the test pixel to the second group of pixels.

29. The method of claim 27 wherein selectively weighting second group pixel values includes weighting in response to number of pixels in the second group.

30. The method of claim 16 wherein generating FV includes:

calculating nV = sum of second group pixel values for pixels having a map value of 0;
calculating nE = total number of pixels in the second group with a map value of 0;
if $nE = 1$, then $FV = (nV + P(i,j) + 1) >> 1$;
if $nE < 4$, then

$$nV = nV + (4 - nE) * P(i,j);$$

and,

$$FV = (nV + 2) >> 2;$$

if $nE < 8$, then

$$nV = nV + (8 - nE) * P(i,j);$$

and,

$$FV = (nV + 4) >> 3;$$

if nE = 8, then

$$nV = nV - P(i + 1, j +1) +P(i,j);$$

and,

$$FV = (nV + 4) >> 3.$$

31. The method of claim 16 wherein generating FV includes:

calculating nV = sum of second group pixel values for pixels having a map value of 0;
calculating nE = total number pixels in the second group with a map value of 0;
if nE = 1, then

$$FV = (nV + P(i,j) +1) >> 1;$$

if nE < 4, then

$$nV = nV + (4 - nE)*P(i,j);$$

and,

$$FV = (nV) >> 2;$$

if nE < 8, then

$$nV = nV + (8 - nE)*P(i,j);$$

and,

$$FV = (nV) >> 3;$$

if nE = 8, then

$$nV = nV - P(i + 1, j +1) +P(i,j);$$

and,

$$FV = (nV) >> 3.$$

32. The method of claim 16 wherein generating a FV in response to the second group data includes:

loading a lookup table (LUT) with the pre-calculated values; and,
accessing the LUT.

33. The method of claim 32 generating a FV includes:

calculating nE = the total number of pixels in the second group with the first map value; and,
using nE to access the LUT.

34. The method of claim 32 loading a LUT with the pre-calculated values includes loading a value for each nE indicating the number of times the test pixel P(i, j) is added.

35. The method of claim 32 loading a LUT with the pre-calculated values includes loading a value for each nE indicating the number of times the result is right shifted.

36. The method of claim 32 loading a LUT with the pre-calculated values includes loading a value for each nE indicating if a pixel value from second group of pixels is subtracted.

37. An image de-ringing filter method, the method comprising:

accepting a plurality of image pixels;
performing a mathematical operation on a first group of pixels neighboring a test pixel;
in response to the first group operation, deciding if the test pixel includes image ringing artifacts;
performing a mathematical operation on a second group of pixels neighboring the test pixel;
in response to the second group operation, generating a filtered value (FV); and,
replacing the test pixel actual value with FV.

38. The method of claim 37 wherein performing a mathematical operation on the first group of pixels includes:

defining a matrix; and,
multiplying the first group of pixels by the matrix.

39. The method of claim 37 wherein performing a mathematical operation on the first group of pixels includes comparing values of pixels on opposite sides of a coordinate axis bisecting the test pixel.

40. The method of claim 37 generating a FV in response to the second group operation includes:

generating a map value for each pixel in the second group; and,
using pixels from the second group to calculate FV, if they are equal to a first map value.

41. An image de-ringing filter system, the system comprising:

a decision unit (102) having an input to accept a plurality of image pixels, the decision unit (102) collecting data from a first group of pixels neighboring a test pixel and, in response to the first group data, supplying a decision at an output as to whether the test pixel includes image ringing artifacts; and,
a filter (108) having an input to accept the plurality of image pixels and an input to accept the decision, the filter collecting data from a second group of pixels neighboring the test pixel and, in response to the second group data, generating a filtered value (FV) and supplying the FV at an output as a replacement to the test pixel actual value.

42. The system of claim 41 wherein the filter (108) performs a mathematical operation on the second group of pixels.

43. The system of claim 41 wherein the decision unit (102) performs a mathematical operation on the first group of pixels.

44. The system of claim 43 wherein the decision unit (102) compares the results of the mathematical operation to a threshold.

45. The system of claim 43 wherein the decision unit (102) defines a matrix and multiplies the first group of pixels by the matrix.

46. The system of claim 45 wherein the decision unit (102) defines a matrix such that a zero value is assigned to the position of the test pixel in the matrix.

47. The system of claim 43 wherein the decision unit (102) compares values of pixels on opposite sides of a coordinate axis bisecting the test pixel.

48. The system of claim 47 wherein the decision unit (102) subtracts the values of pixels on a first side of the coordinate axis from pixels on a second side of the coordinate axis, opposite of the first side, and compares the difference to a threshold.

49. The system of claim 48 wherein the decision unit (102) compares the values of pixels on opposite sides of a plurality of coordinate axes, oriented in a plurality of directions.

50. The system of claim 47 wherein the decision unit (102) collects data from a group of 4 pixels neighboring the test pixel.

51. The system of claim 49 wherein the decision unit (102) :

with respect to a test pixel P(i,j), with i and j indicating row and column indices, respectively, and P(i,j) represents a pixel gray value, uses operators H1 and H2 to derive gradient values $g_{H1}(i,j)$ and $g_{H2}(i,j)$, respectively, where

$$H1 = [10\ -1];$$

and,

$$H2=\begin{bmatrix}1\\0\\-1\end{bmatrix}$$

calculates

$$S(i,j) = (|g_{H1}(i,j)| + |g_{H2}(i,j)| + 1) >> 1;$$

where >> x represents a binary value right-shift of x; and,
decides that P(i,j) is a ringing artifact, if S(i,j) < threshold.

52. The system of claim 41 further comprising:

a decoder (112) having a connection to accepted encoded video information and an output to supply the plurality of image pixels to the decision unit (102), and to the filter (108), as decoded image information.

53. The system of claim 41 wherein the filter (108) adds the test pixel to the second group of pixels.

54. The system of claim 41 wherein the filter (108) collects data from 8 pixels neighboring the test pixel.

55. The system of claim 51 wherein the decision unit (102) :

in response to comparing S(i,j) to the threshold, generates a map value M(i,j) for P(i,j), where:

$$M(i,j) = 1, \text{ if } S(i,j) \geq \text{threshold};$$

and,

$$M(i,j) = 0, \text{ if } S(i,j) < \text{threshold};$$

wherein the filter (108) generates a map value for each pixel in the second group and uses pixels from the second group to calculate FV, if they are equal to a first map value.

56. The system of claim 55 wherein the filter (108) uses a first map value equal to 0.

57. The system of claim 55 wherein the filter (108) uses a first map value not equal to 1.

58. The system of claim 55 wherein the filter (108) selects pixels from the second group to calculate FV, if they are equal to the first map value.

59. The system of claim 58 wherein the filter (108) randomly selects pixels from the second group to calculate FV, if they are equal to the first map value.

60. The system of claim 59 wherein the filter (108) accepts a plurality of image pixel sets, in a plurality of frames and generates FV by:

randomly selecting a first collection of pixel positions with respect to the test pixel;
using pixels in the first collection to calculate FV for each test pixel in every image pixel set, in every frame.

61. The system of claim 59 wherein the filter (108) accepts a plurality of image pixel sets, in a plurality of frames and generates FV by:

randomly selecting a first collection of pixel positions with respect to the test pixel in a first image pixel set in a current frame;
using pixels in the first collection to calculate FV for each test pixel in every image pixel set in the current frame;
randomly reselecting a second collection of pixel positions in an image pixel set in a frame subsequent to the current frame; and,
using pixels in the second collection to calculate FV for each test pixel in every image pixel set in the subsequent frame.

62. The system of claim 58 wherein the filter (108) selects predetermined pixel positions, with respect to the test pixel, from the second group to calculate FV, if they are equal to the first map value.

63. The system of claim 62 wherein the filter (108) accepts a plurality of image pixel sets in a plurality of frames and selects the pixels in the predetermined pixel positions to calculate FV for each test pixel in every image pixel set, in every frame.

64. The system of claim 62 wherein the filter (108) accepts a plurality of image pixel sets in a plurality of frames and generates FV by:

selecting the pixels in a predetermined first collection of pixel positions to calculate FV for each test pixel in every image pixel set in a current frame; and,
selecting the pixels in a predetermined second collection of pixel positions to calculate FV for each test pixel in every image pixel set in a frame subsequent to the current frame.

65. The system of claim 62 wherein the filter (108) generates FV by:

selecting the pixels in a predetermined first collection of pixel positions to calculate FV for test pixels in a first image pixel set; and,
selecting the pixels in a predetermined second collection of pixel positions to calculate FV for test pixels a second image pixel set.

66. The system of claim 55 wherein the filter (108) uses pixels from the second group to calculate FV, if they are equal to a first map value, by:

selectively weighting second group pixel values;
summing the weighted values; and,
averaging.

67. The system of claim 66 wherein the filter (108) adds the test pixel to the second group of pixels.

68. The system of claim 66 wherein the filter (108) selectively weights in response to number of pixels in the second group.

69. The system of claim 55 wherein the filter (108) generates FV by:

calculating nV = sum of second group pixel values for pixels having a map value of 0;
calculating nE = total number of pixels in the second group with a map value of 0;
if nE = 1, then

$$FV = (nV + P(i,j) + 1) >> 1;$$

if nE < 4, then

$$nV = nV + (4 - nE) * P(i,j);$$

and,

FV=(nV+2)>>2;

if nE < 8, then

nV = nV + (8 - nE)∗P(i,j);

and,

FV = (nV + 4) >> 3;

ifnE = 8,then

nV = nV - P(i + 1, j +1) +P(i,j);

and,

FV=(nV+4)>>3.

70. The system of claim 55 wherein the filter (108) generates FV by:

calculating nV = sum of second group pixel values for pixels having a map value of 0;
calculating nE = total number pixels in the second group with a map value of 0;
if nE = 1, then

FV = (nV + P(i,j) +1) >> 1;

if nE < 4, then

nV = nV + (4 - nE)∗P(i,j);

and,

FV = (nV) >> 2;

if nE < 8, then

nV = nV + (8 - nE)∗P(i,j);

and,

FV = (nV) >> 3;

if nE = 8, then

nV = nV - P(i + 1, j +1) +P(i,j);

and,

FV (nV) >> 3.

71. The system of claim 55 further comprising:

an accessible memory (120) including a lookup table (LUT) (122) with pre-calculated values;

wherein the filter (108) generates a FV in response accessing the LUT (122).

72. The system of claim 71 wherein the LUT (122) is indexed by nE values: and,
wherein the filter (108) calculates nE = the total number of pixels in the second group with the first map value, and uses nE to access the LUT (122).

73. The system of claim 72 wherein the LUT (122) includes a value for each nE indicating the number of times the test pixel P(i,j) is added.

74. The system of claim 72 wherein the LUT (122) includes a value for each nE indicating the number of times the result is right shifted.

75. The system of claim 72 wherein the LUT (122) includes a value for each nE indicating if a pixel value from second group of pixels is subtracted.

FIG.1

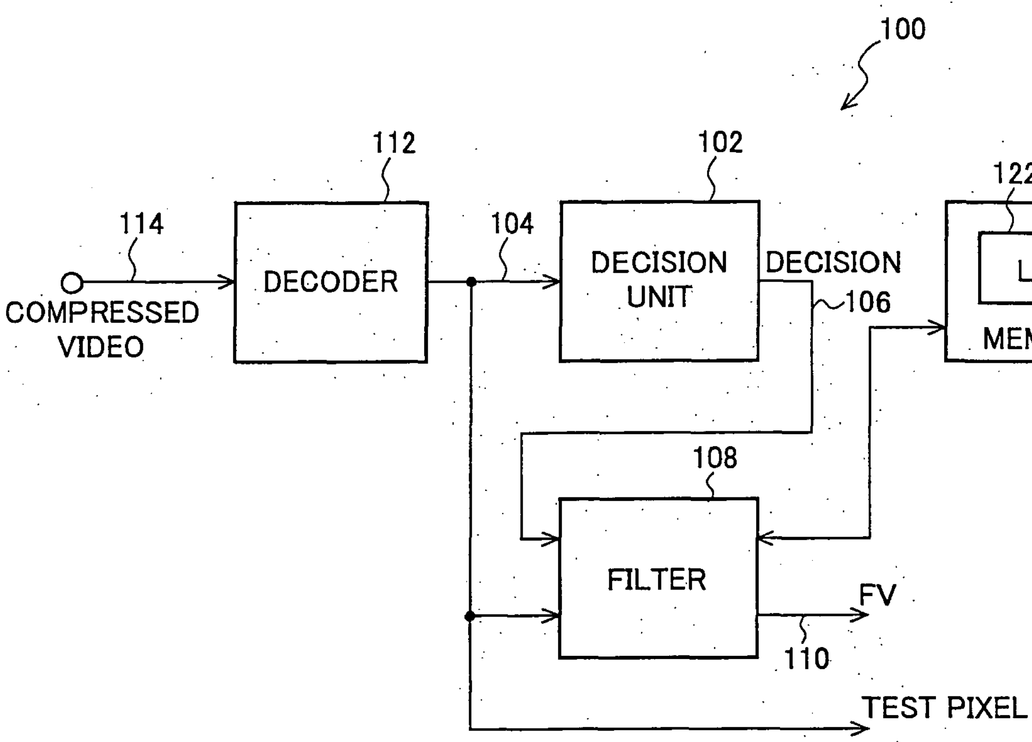

100
112
102
120
122
114
104
DECODER
DECISION UNIT
DECISION
106
LUT
MEMORY
COMPRESSED VIDEO
108
FILTER
FV
110
TEST PIXEL

FIG.2

1a 2a	1b 2b	1c 2c
1d 2d	1e TEST PIXEL 2e	1f 2f
1g 2g	1h 2h	1i 2i

FIG.3

nE	TIMES ADDED	TIMES SUBTRACTED	TIMES RIGHT-SHIFTED	
1	1	0	1	
2	2	0	2	
3	1	0	2	
4	4	0	3	
5	3	0	3	
6	2	0	3	
7	1	0	3	
8	1	1	3	

122

LUT

FIG.4

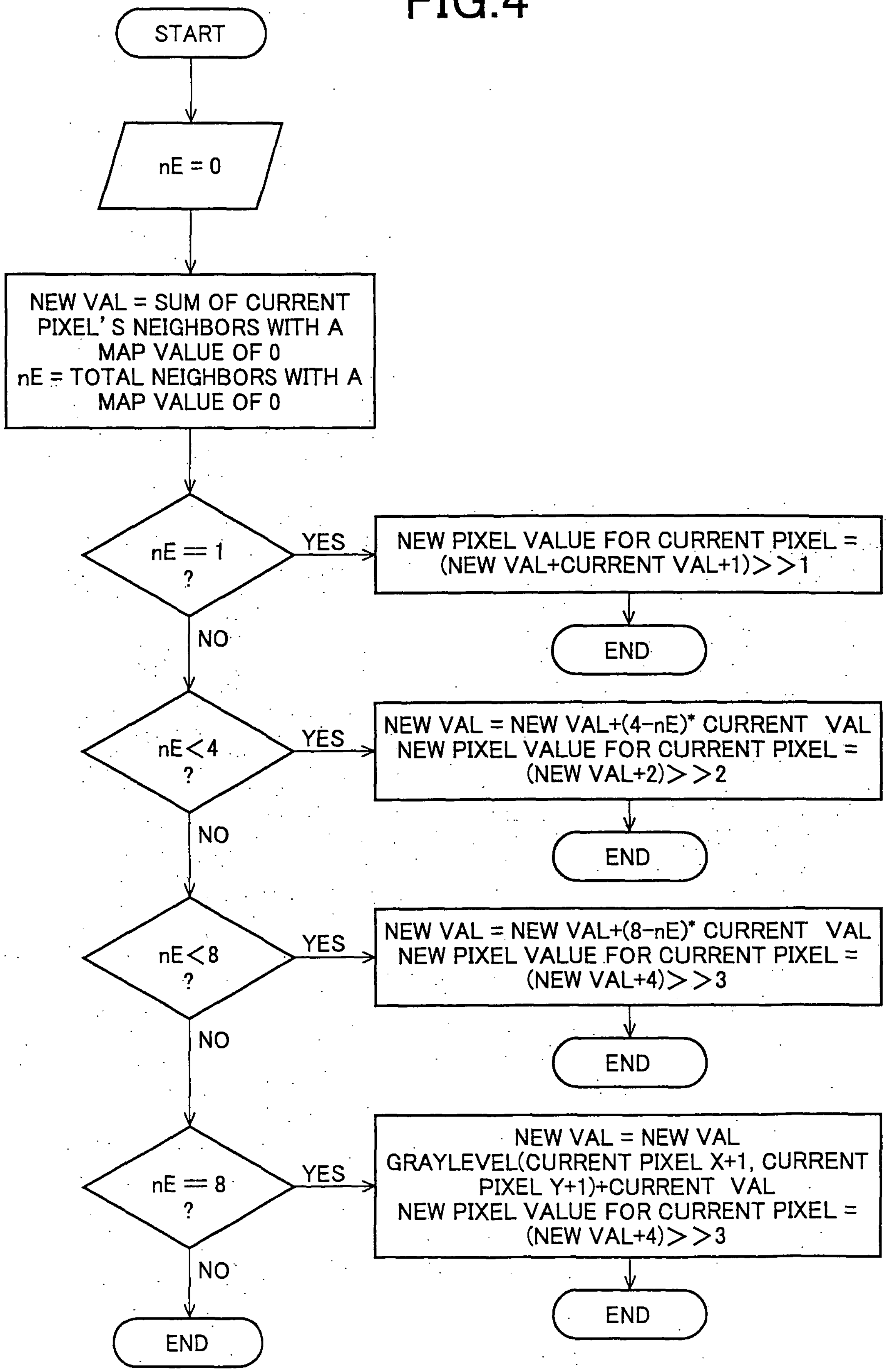

START
nE = 0
NEW VAL = SUM OF CURRENT PIXEL' S NEIGHBORS WITH A MAP VALUE OF 0
nE = TOTAL NEIGHBORS WITH A MAP VALUE OF 0
nE == 1 ?
YES
NEW PIXEL VALUE FOR CURRENT PIXEL = (NEW VAL+CURRENT VAL+1)>>1
END
NO
nE<4 ?
YES
NEW VAL = NEW VAL+(4-nE)* CURRENT VAL
NEW PIXEL VALUE FOR CURRENT PIXEL = (NEW VAL+2)>>2
END
NO
nE<8 ?
YES
NEW VAL = NEW VAL+(8-nE)* CURRENT VAL
NEW PIXEL VALUE FOR CURRENT PIXEL = (NEW VAL+4)>>3
END
NO
nE == 8 ?
YES
NEW VAL = NEW VAL
GRAYLEVEL(CURRENT PIXEL X+1, CURRENT PIXEL Y+1)+CURRENT VAL
NEW PIXEL VALUE FOR CURRENT PIXEL = (NEW VAL+4)>>3
END
NO
END

FIG.5

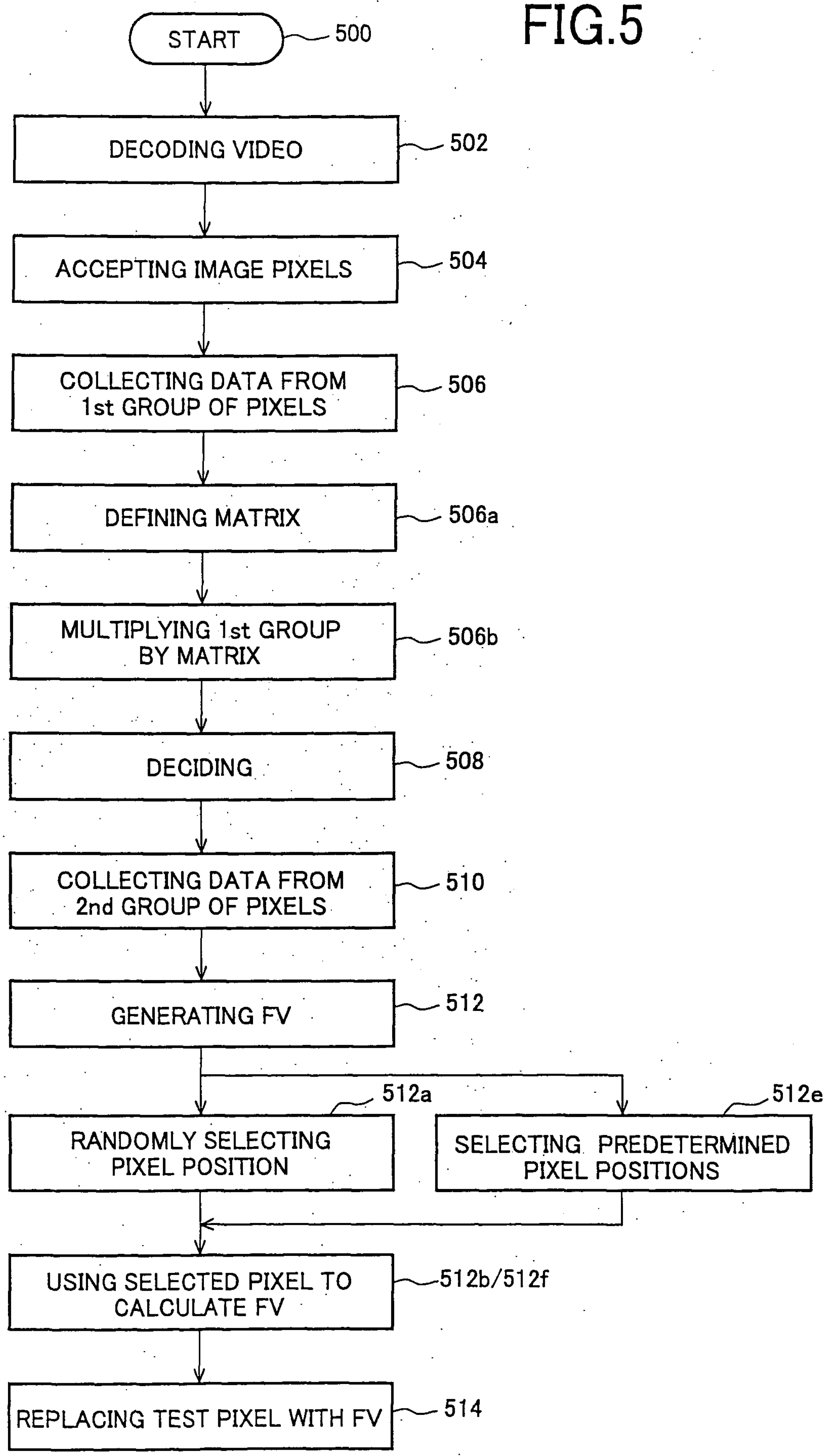
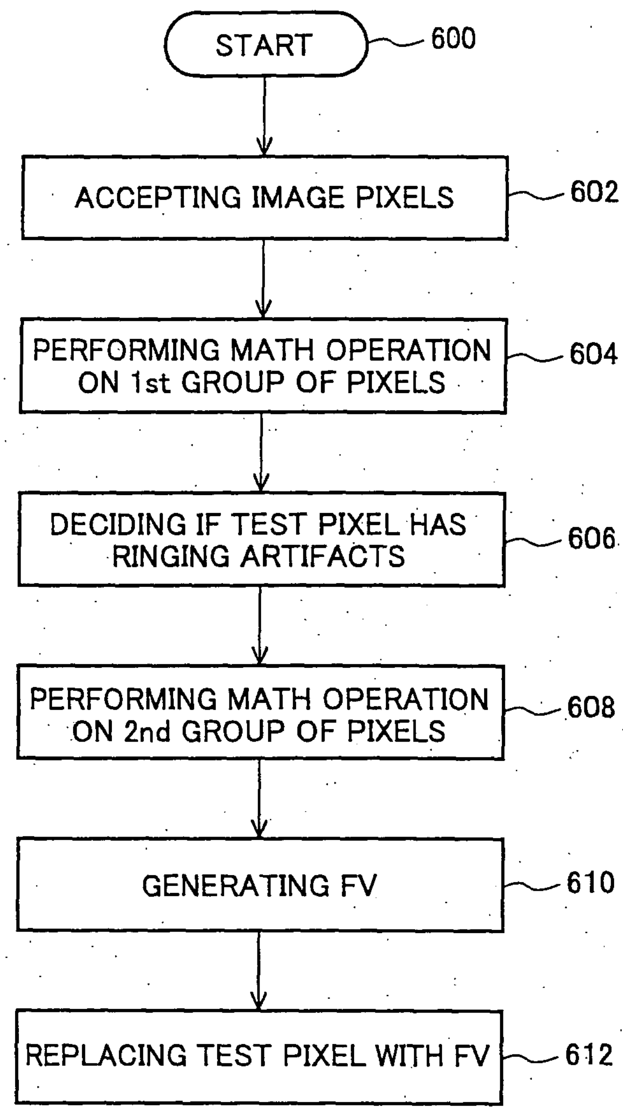

START
500
DECODING VIDEO
502
ACCEPTING IMAGE PIXELS
504
COLLECTING DATA FROM 1st GROUP OF PIXELS
506
DEFINING MATRIX
506a
MULTIPLYING 1st GROUP BY MATRIX
506b
DECIDING
508
COLLECTING DATA FROM 2nd GROUP OF PIXELS
510
GENERATING FV
512
512a
RANDOMLY SELECTING PIXEL POSITION
512e
SELECTING PREDETERMINED PIXEL POSITIONS
USING SELECTED PIXEL TO CALCULATE FV
512b/512f
REPLACING TEST PIXEL WITH FV
514

FIG.6


START
600
ACCEPTING IMAGE PIXELS
602
PERFORMING MATH OPERATION ON 1st GROUP OF PIXELS
604
DECIDING IF TEST PIXEL HAS RINGING ARTIFACTS
606
PERFORMING MATH OPERATION ON 2nd GROUP OF PIXELS
608
GENERATING FV
610
REPLACING TEST PIXEL WITH FV
612

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 04 03 0974

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	US 2003/081854 A1 (DESHPANDE SACHIN G) 1 May 2003 (2003-05-01)	1-5,10, 12-15, 37,38, 41-45, 50,52-54	H04N7/26 G06T9/00 H04N7/30 G06T5/00
Y	* abstract; claims 1-12; figures 1-5 *	11, 16-36, 40,51, 55-75	
	* paragraphs [0034] - [0067] *		
X	US 2002/122602 A1 (ROUVELLOU LAURENT JEAN MARIE) 5 September 2002 (2002-09-05)	1-9,12, 37-39, 41-49	
Y	* abstract; claims 1-3; figure 1 *	11, 16-36, 40,51, 55-75	
	* paragraphs [0006] - [0013], [0028] - [0055] *		
X	EP 1 351 508 A (SEIKO EPSON CORPORATION) 8 October 2003 (2003-10-08) * abstract; figures 2,4 * * paragraphs [0006] - [0011], [0015] - [0018] *	1,37,41	TECHNICAL FIELDS SEARCHED (Int.Cl.7) H04N G06T
X	SKOCIR P ET AL: "Spatio-temporal post-filtering of 3-D coded video sequences" ELECTROTECHNICAL CONFERENCE, 2000. MELECON 2000. 10TH MEDITERRANEAN MAY 29-31, 2000, PISCATAWAY, NJ, USA,IEEE, vol. 2, 29 May 2000 (2000-05-29), pages 474-477, XP010517831 ISBN: 0-7803-6290-X * abstract * * paragraphs [01.B], [0003] *	1,37,41	
	-/--		

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
Munich	23 February 2005	Cakiroglu, S

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

4

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 04 03 0974

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	HU J ET AL: "Removal of blocking and ringing artifacts in transform coded images" ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, 1997. ICASSP-97., 1997 IEEE INTERNATIONAL CONFERENCE ON MUNICH, GERMANY 21-24 APRIL 1997, LOS ALAMITOS, CA, USA,IEEE COMPUT. SOC, US, vol. 4, 21 April 1997 (1997-04-21), pages 2565-2568, XP010225679 ISBN: 0-8186-7919-0 * abstract * * paragraphs [02.1], [02.3] * -----	1,37,41	
A	WO 00/14968 A (WISCONSIN ALUMNI RESEARCH FOUNDATION; HU, YU, HEN; NGUYEN, TRUONG, Q;) 16 March 2000 (2000-03-16) * abstract; figures 2,4,5,8,9,12 * * page 2, line 11 - page 3, line 21 * * page 5, line 30 - page 11, line 4 * -----	1-75	
			TECHNICAL FIELDS SEARCHED (Int.Cl.7)
A	WO 03/081534 A (KONINKLIJKE PHILIPS ELECTRONICS N.V; GESNOT, ARNAUD) 2 October 2003 (2003-10-02) * abstract * * page 2, line 19 - page 3, line 11 * * page 4, line 10 - page 6, line 2; claim 1 * -----	1-75	
A	EP 0 797 349 A (SAMSUNG ELECTRONICS CO., LTD) 24 September 1997 (1997-09-24) * abstract * * page 2, lines 3-35 * * page 3, line 3 - page 4, line 42; claim 1; figure 1 * -----	1-75	

The present search report has been drawn up for all claims

4

Place of search	Date of completion of the search	Examiner
Munich	23 February 2005	Cakiroglu, S

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons
& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 04 03 0974

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

23-02-2005

Patent document cited in search report		Publication date	Patent family member(s)			Publication date
US 2003081854	A1	01-05-2003	NONE			
US 2002122602	A1	05-09-2002	FR	2818863	A1	28-06-2002
			CN	1362832	A	07-08-2002
			EP	1223764	A1	17-07-2002
			JP	2002247414	A	30-08-2002
EP 1351508	A	08-10-2003	US	2003190086	A1	09-10-2003
			CN	1450811	A	22-10-2003
			EP	1351508	A2	08-10-2003
			JP	2003339048	A	28-11-2003
WO 0014968	A	16-03-2000	AU	5822299	A	27-03-2000
			WO	0014968	A1	16-03-2000
			US	6668097	B1	23-12-2003
WO 03081534	A	02-10-2003	AU	2003208527	A1	08-10-2003
			EP	1490835	A1	29-12-2004
			WO	03081534	A1	02-10-2003
EP 0797349	A	24-09-1997	KR	242636	B1	01-02-2000
			CN	1170304	A ,C	14-01-1998
			DE	69627982	D1	12-06-2003
			DE	69627982	T2	11-03-2004
			EP	0797349	A2	24-09-1997
			US	5883983	A	16-03-1999

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82